



دانشگاه صنعتی شریف

تمرین سوم درس متدولوژی های ایجاد نرم افزار

استاد:

دکتر رامسین

تهیه کننده:

امید یعقوبی سامانی

۴۰۱۲۰۵۳۴۸

فهرست مطالب

۱	مقایسه متدولوژی DSDM با متدولوژی Crystal Clear
۸	معیارهای فرایند:
۸	معیار تعریف:
۱۱	معیار پوشش چرخه عمر عمومی ایجاد نرم افزار:
۱۴	معیار پشتیبانی از فعالیت های چتری:
۱۶	معیار بی درزی و گذر هموار:
۱۶	معیار مبتنی بودن بر نیازمندی ها:
۱۷	معیار آزمون پذیری، ملموس بودن و قابل رهگیری بودن به نیازمندی ها:
۱۸	معیار تشویق به مشارکت فعالانه کاربران:
۱۸	معیار قابل اجرا بودن و کارا بودن:
۱۹	معیار قابل مدیریت بودن پیچیدگی:
۲۰	معیار قابلیت گسترش، مقیاس پذیری، پیکربندی و انعطاف پذیری:
۲۱	معیار حوزه کاربرد:
۲۲	معیارهای زبان مدل سازی:
۲۲	معیار پشتیبانی از مدل سازی شی گرا سازگار، دقیق و بدون ابهام:
۲۳	معیار مدیریت تناقض ها و پیچیدگی ها:

مقایسه متدولوژی DSDM با متدولوژی Crystal Clear:

متدولوژی DSDM

DSDM یا Dynamic Systems Development Method یک متدولوژی چابک است. رویکرد DSDM هم در فرآیند ایجاد و هم تحویل، تکراری و افزایشی است و نیازهای کسب و کاری مهم خیلی زود و در ابتدای کار و نیازهای کمتر مهم در انتها محقق می شوند. رویکرد تکراری DSDM نمایندگان کسب و کار را قادر می سازد که راه حل را به صورت تکاملی ببینند و بازخورد بدهند و درخواست تغییرات را در طول مدت ایجاد راه حل ارسال کنند. برخلاف بیشتر رویکردهای چابک در DSDM مدیریت پروژه و ایجاد محصول در یک فرآیند قرارداد شده اند و برای بیشتر سازمان ها DSDM کاملاً نیازهایشان را برآورده می کند.

فرآیند پروژه دارای چهار فاز اصلی Feasibility، Foundation، Evolutionary Development و Deployment است که قبل اینها فاز Pre-Project و بعد اینها Post-Project است و در مجموع ۶ فاز دارد.

۱. فاز Pre-Project:

در این فاز اطمینان حاصل می شود که پروژه درست آغاز شده و به طور درستی هم آغاز شده و بر پایه تعریف دقیق اهداف است.

۲. فاز Feasibility:

در این فاز ابتدا بررسی می شود آیا پروژه از دیدگاه فنی شدنی است یا خیر و از دیدگاه کسب و کار توجیه اقتصادی دارد یا نه. میزان تلاش انجام شده در این فاز باید به اندازه ای باشد که بتواند توجیه کننده بررسی های آینده باشد یا اینکه باعث شود پروژه همین جا متوقف شود.

۳. فاز Foundation:

این فاز بررسی های اولیه در فاز امکان سنجی را به سطح بعدی می برد. هدف این فاز ایجاد یک فهم بنیادی از منطق کسب و کار پروژه و نحوه مدیریت ایجاد و تحویل راه حل است. سطح جزئیات این فاز نباید خیلی بالا باشد و این فاز حتی برای پروژه های خیلی بزرگ نباید بیش از چند هفته طول بکشد.

جزئیات نیازمندی ها و چگونگی تحقق آنها به عنوان بخشی از راه حل در فاز ایجاد تکاملی انجام خواهد شد. گاهی لازم می شود بعد از فاز استقرار به این فاز برگشت. تصمیم برای بازگشت می

تواند در آغاز پروژه برنامه ریزی شود. برای مثال در پروژه هایی که محیط کسب و کار پویا است ممکن است باعث بوجود آمدن تغییرات زیادی در این فاز شوند. در غیر این صورت تصمیم به بازگشت در انتهای استقرار و زمانی که نتیجه مناسب نباشد اتخاذ می شود. هدف اصلی این فاز فهم حوزه کاری، چگونگی انجام آن و اینکه توسط چه کسی، چه زمانی و در کجا انجام شود است. این فاز همچنین چرخه حیات پروژه را بوسیله توافق روی چگونگی اعمال فرآیند DSDM روی نیازهای خاصی از پروژه تعیین می کند.

۴. فاز Evolutionary Development:

هدف این فاز ساخت پروژه به صورت تکاملی روی زیرساخت مبنایی که در فاز قبل برای پروژه مشخص شده است. در این فاز نیاز است تیم ایجاد راه حل، رویه هایی مثل ایجاد تکراری، timebox بودن، اولویت MoSCoW را به همراه مدل سازی و کارگاه تسهیل گری اعمال کند تا راه حل دقیق را در طول مدت پروژه طوری پوشش بدهد که نیازهای کسب و کار را محقق کند و آن را از نقطه نظر درست فنی بسازد.

در این فاز کارها Timebox انجام می شود، تیم ایجاد راه حل را به صورت افزایشی می سازد، به طور تکراری نیازمندی های سطح پایین کاوش می شوند و تست به طور مداوم انجام می شود.

۵. فاز Deployment:

هدف در این فاز ساخت یک خط مبنا برای بردن راه حل تکاملی به محیط عملیاتی است. عرضه ای که استقرار می یابد ممکن است راه حل نهایی یا بخشی از راه حل نهایی باشد. این فاز از سه فعالیت اصلی تشکیل شده است: مونتاژ، بازبینی و استقرار. به علاوه بعد از آخرین عرضه پروژه به طور رسمی بسته خواهد شد.

آ. مونتاژ: قبل از استقرار فیزیکی معمولاً فعالیت هایی برای اطمینان از اینکه چیزی که تحویل داده می شود همراستا و منسجم است انجام می شود. همچنین می تواند شامل جمع آوری اطلاعات مربوط به پشتیبانی باشد. این فعالیت شامل جمع آوری قطعاتی که باید عرضه شوند است. در پروژه های ساده کوچک کارها در این فعالیت کمینه است. در پروژه های پیچیده تر که چند پروژه برای عرضه باید با هم یکی شوند کار انجام شده برای مونتاژ قطعات قابل توجه است و مواردی مثل ترکیب فرآیند های جدید کسب و کار، زمان بندی برای آموزش، راهنمای کاربر و راه حل جدید فناوری اطلاعات را شامل می شود.

ب. فعالیت بازبینی: وقتی همه عناصری که باید عرضه شوند مونتاز شدند در بیشتر شرایط نوعی تاییدیه استقرار وجود دارد که بر پایه بازبینی نهایی راه حل قبل از رفتن به محیط عملیاتی است تا اطمینان حاصل شود عرضه دارای استانداردهای مناسب است و به اندازه کافی کامل می باشد. در پروژه های ساده این کار را می توان به طور غیر رسمی و با یک چک لیست ساده و در پروژه های پیچیده می تواند رسمی و به صورت کارگاه باشد. در این نقطه تیم، **retrospective** و بازبینی رسمی محصول را با تمرکز روی فرآیند و نواحی که می تواند بهبود یابد انجام می دهد. برنامه ریزی برای فرآورده های بعدی انجام می شود و می تواند به تسهیل یادگیری در طی پروژه کمک کند.

ج. فعالیت استقرار: بعد از تایید نهایی، استقرار که عمل فیزیکی انتقال موارد مونتاز شده به محیط عملیاتی است انجام می شود که شامل کارهای فنی مثل انتقال راه حل به محیط کاربر و همچنین برنامه برای تغییرات محیط کسب کار که باید اعمال شود است.

بعد از استقرار نهایی پروژه به طور رسمی بسته می شود. در این زمان کل تیم یک **retrospective** برای بازبینی کارایی کل پروژه برگزار می کند که شامل بازبینی تکنیکی و فرآیندی و بازبینی از دید کسب و کار است.

۶. فاز Post-Project:

بعد از استقرار نهایی پروژه، در این فاز بررسی می شود که منافع مورد انتظار پروژه برآورده شده باشد. ممکن است برخی منافع برجسته شوند و اغلب مزایای راه حل در طول دوره ای تعریف شده محقق شوند. در این فاز یک یا چند ارزیابی برای منافع تحقق نیافته ای که در رابطه با توجیه اقتصادی هستند انجام می شود. بسته به نیازهای سازمان ارزیابی می تواند برای عرضه های تکی یا کل پروژه باشد.

متدولوژی Crystal Clear

چرخه پروژه متدولوژی Crystal Clear دارای سه بخش است:

۱. Chartering :

تکالیف در این فاز از چند هفته تا چند روز طول می کشد و شامل موارد زیر است:

آ. ساخت هسته تیم:

پروژه هایی که از **crystal clear** استفاده می کنند تیم اغلب با یک اسپانسر اجرایی، یک رهبر طراح و یک کاربر کلیدی ایجاد می شود. به طور معمول دو تا پنج شخص دیگر هم به پروژه اضافه خواهند

شد که ترکیب های مختلفی از مهارت، تجربه و توانایی را دارند. هر چقدر تیم تازه کارتر باشد برای رهبر طراح انجام هر دو کار آموزش و ایجاد نرم افزار سخت تر خواهد بود. پس اگر بیشتر از چهار نفر درگیر کار باشند تیم باید شامل حداقل یک فرد باتجربه و شایسته باشد تا بتواند از رهبر پشتیبانی و بار کاری اش را کم کند. هر دو یا سه شخص تازه کار نیازمند یک شخص با تجربه و شایسته است.

ب. اجرای کاوشی ۳۶۰ درجه:

در یک نقطه ی آغازین پروژه تیم شامل اسپانسر اجرایی یکبار روی مسائل بررسی انجام می دهد. این کار کاوشی ۳۶۰ درجه است و معادل فاز Inception در متدولوژی RUP است. در این بررسی ارزش تجاری، نیازمندی ها، مدل قلمرو، تکنولوژی، نوع پروژه، آرایش تیم و متدولوژی یا قراردادهایی که باید استفاده شوند بررسی می شوند. هر کدام از اینها به صورت درشت دانه و سطح بالا بررسی می شوند تا مشخص کند چگونه سیستم مورد نظر و متدولوژی پیش بینی شده از تکنولوژی مورد نظر حول مدل قلمرو استفاده می کنند و می توانند ارزش تجاری مورد نظر را با مجموعه پیش بینی شده نیازمندی ها طبق پیش نویس برنامه پروژه محقق کند. نتیجه ۳۶۰ درجه کاوشی مجموعه تنظیمات برای آغاز پروژه یا در موارد جدی تر تصمیم اسپانسر اجرایی برای لغو پروژه است.

پ. شکل دهی و تنظیم دقیق قراردادهای متدولوژی:

شکل دادن متدولوژی می تواند اغلب در دو روز انجام بشود و برای تیم crystal clear نباید از هفته بیشتر بشود. این کار می تواند با تکنیک های شکل دهی متدولوژی انجام بشود. متدولوژی چیزی بیش از قراردادهایی که تیم روی آن توافق دارد نیست. چون مجموعه قراردادها دو بار در هر تکرار یا چرخه تحویل مورد بازبینی قرار می گیرند ممکن است با هر متدولوژی شروع و به طور ساده ای آن را در طول زمان طوری تنظیم کرد که مناسب تیم باشد. اما احتمالاً برای تیم crystal clear زمان کافی برای پیروی از این استراتژی وجود ندارد پس سریع ترین کار این است با یک لیست فکر شده توسط تیم شروع شود که عواقب آن شدید و جدی نباشد. بهتر است با مجموعه اولیه قراردادها که می دانیم تقریباً درست هستند شروع شود و چون معمولاً تیم کار نرم افزاری بیشتری دارد می توان با بررسی قراردادها به طور تجربی و در عمل در reflection workshop های بعدی به این قراردادها مواردی را اضافه نماید.

ت. ساخت برنامه اولیه پروژه:

روش های مختلفی در محدودیت های crystal clear برای ساخت برنامه پایه پروژه وجود دارد. تکنیک Blitz Planning، روش های دقیق شناخته شده در DSDM که شامل مجموعه ای از بررسی های اهداف پروژه توسط اسپانسر اجرایی است و در ادامه ساخت برنامه عرضه و مجموعه ای timebox ها که هر کدام یک تا سه ماه طول می کشند. از روش های برنامه ریزی اسکرام و برنامه ریزی xp هم می توان استفاده نمود. هر کدام از این تکنیک ها در محدوده تحمل متدولوژی crystal clear قرار دارد. هر کدام برنامه اولیه ارائه می دهند و به اندازه کافی سریع هستند که بتوانند از یک اساس با قاعده برای رهگیری تغییرات در موفقیت پروژه استفاده کنند و برنامه پروژه را بروز رسانی کنند.

برای پروژه های متوسط تیم در طی دوره chartering ایجاد می شود. افرادی که chartering را انجام می دهند با تعدادی متغیر سر و کار دارند. افراد ممکن است همدیگر را بشناسند یا نشناسند و یا ممکن است در هر نقطه ای در میانه پروژه ظاهر شوند که نیازمند زمان برای یادگیری ارتباط با هم هستند.

اگر نتیجه کاوشی ۳۶۰ درجه مثبت باشد تیم متوجه می شود پروژه به کجا می رسد، چه مشکلی رفع می شود، چه تکنولوژی استفاده می شود و برنامه پروژه و ارزش تجاری چیست و یک تنظیم اولیه از اهداف و اعمالی که تیم را قوی می کند ساخته خواهد شد.

شکل دادن متدولوژی و برنامه ریزی اولیه پروژه چند روز طول می کشد. اگر نیازمند ارتباط با افراد بیشتری باشد ممکن است چند روز بیشتر هم طول بکشد.

۲. Delivery Cycle:

چرخه تحویل چهار بخش دارد:

آ. تنظیم مجدد برنامه عرضه:

در اولین چرخه تحویل، برنامه ساخته شده مجدداً تنظیم می شود که البته ضروری نیست. بعد از اولین تحویل اعضای تیم، اطلاعات جدید و با ارزشی برای استفاده در برنامه پروژه پیدا می کنند. همچنین هم یاد می گیرند چقدر واقعا سریع کار می کنند و چطور تخمین های اولیه آنها اشتباه بوده است. به علاوه تیم و کاربران بیشتر در مورد آنچه واقعا در سیستم نیاز است می فهمند.

دو راه وجود دارد یا به نیازمندی ها و برنامه اولیه پایبند بود یا هم نیازمندی ها و برنامه را بازبینی کرد. در crystal clear اسپانسر پروژه دستور نمی دهد و تیم اگر برنامه و تخمین ها را بروزرسانی نکند خودش بی مبالاتی کرده است.

ب. تکرارها:

طول تکرار و شکل آن در تیم های مختلف متفاوت است دو نوع چرخه تکرار ممکن است وجود داشته باشد. تکرار یک هفته ای و تکرار های دو ماهه. می توان بین این دو را هم انتخاب کرد. هر تکرار سه بخش دارد:

- برنامه ریزی تکرار
- فعالیت روزانه و چرخه ادغام
- مراسم پایانی

در دوره تکرار، تیم به مجموعه نیازمندی های اضافه می کند، رابط کاربری را طراحی می کند، زیرساخت سیستم را گسترش می دهد، عملکرد اضافه می کند، سیستم را به کاربران نشان می دهد، تست اضافه می کند و آنها را به قابلیت های خودکار محیط پروژه اضافه می کند.

چرخه ادغام می تواند از نیم ساعت تا چندین روز بسته به تجربه تیم و رویه هایی که به کار می برد طول بکشد. اگرچه هر چه کوتاه تر باشد بهتر است اما متدولوژی درباره طول اجرای ادغام و یکپارچه سازی کد زمانی را تعیین نکرده است.

در پایان تکرارها یک reflection workshop برگزار و یک مراسم پایانی هم برگزار خواهد شد و تکرارها را می سنجد که در صورت موفقیت باعث رضایت افراد خواهد شد و در صورت عدم موفقیت برنامه، محصول و فرایند مورد بازبینی قرار گرفته و علل شکست مشخص بررسی خواهد شد.

پ. تحویل به کاربران واقعی:

تحویل در این متدولوژی به معنای تحویل به کاربر واقعی است. در شرایط مختلف ممکن است استقرار به طور کامل همراه با کلاس آموزشی برای آن باشد یا ممکن است تنها به یک کاربر تحویل داده شود. روش اول گران تر و روش دوم ارزان تر است و در آن می توان از کاربر اطلاعات گران بهایی بدست آورد. سه دلیل وجود دارد که تحویل تا حداکثر هر سه ماه یکبار و نه بیشتر پذیرفته شده است:

- افراد می توانند تمرکز و حافظه را حدود سه ماه حفظ کنند.
- هزینه تحویل سیستم کامل و آموزش به کاربران بسیار بیشتر از تحویل سیستم به یک با چند کاربر است. زمان و هزینه مالی سیستم، نوشتن راهنماها و ساخت آموزش کافی در عمل ممکن نیست که بیش از سه ماه طول بکشد.
- تحویل ارزش تجاری به افراد اجرایی در این محدوده سه ماهه اغلب کافی است.

ت. مراسم پایانی:

زمانی که پایان دوره تحویل نزدیک می شود معمولا فشار خیلی شدید خواهد شد. استراحت بخشی از نیاز انسان برای داشتن ریتم است و افراد می توانند با استراحت خود را برای دوره رشد بعدی آماده کنند. بنابراین بعد از تحویل نرم افزار باید استراحت کرد و **reflection workshop** بخشی از آن است و به جای **reflection workshop** در پایان آخرین تکرار قبل تحویل می توان آن را بعد تحویل انجام داد که شامل مراسم بعد از تحویل هم خواهد بود.

بعد از تحویل، تیم باید روی دو موضوع تفکر کند:

- اینکه استقرار چگونه خواهد شد؟ چه اعمال مختلفی باید در ابتدای چرخه تحویل باید انجام بشود تا از مشکلات استقرار بکاهد؟
- اینکه کاربران در مورد سیستم چه فکر می کنند؟ نقاط ضعف و قوت چیست؟ و مهم تر از آن تیم می تواند درباره نیاز واقعی کاربر در برابر خواسته کاربر یاد بگیرد.

تفکر روی فرایند تحویل مانند هر **reflection workshop** دیگری است. تیم باید ارزیابی شود، تیم باید کاربران را ببیند، باید روی گروهی از کاربران تمرکز کنند، حتی گروهی خارجی را برای ارزیابی سیستم به کار بگیرند. در این سنجش محصول مورد بازبینی قرار می گیرد و نه فرآیند.

۳. Wrap up:

در انتهای کار برای جمع بندی یک تست پذیرش در محیط ایجاد انجام خواهد شد و محیط کاربر آماده استقرار نهایی خواهد شد. یک **reflection workshop** هم برگزار خواهد شد. در تکرارهای یک هفته ای آیین پایانی بیشتر احساسی است تا اینکه در آن کاری انجام بشود. هر چقدر تکرار طولانی تر باشد ادغام طولانی تر خواهد شد به علاوه فرایند های پس از آن هم طولانی تر می شوند و در تکرارهای کوتاه تر نباید انتظار کار خاصی داشت. احتمالا بعد از چند هفته تیم می تواند روی

تغییرات تفکر کند و این مدت کوتاه خواهد بود و آیین پایانی نهایتاً نصف روز طول خواهد کشید. در تکرارهای دو ماهه در نهایت باید تمام جنبه های کار بررسی شود، از ارتباط بین اسپانسر و کاربر تا الگوی ارتباطات بررسی می شود. روش جمع آوری نیازمندی ها و قراردادهای کدنویسی بررسی می شوند. تمرین هایی که انجام نشده یا باید انجام شوند هم بررسی خواهد شد. همچنین تکنیک هایی که باید استفاده شوند هم مشخص خواهند شد.

در ادامه این متدولوژی ها بررسی و مقایسه بین آنها صورت می گیرد.

معیارهای فرایند:

معیار تعریف:

متدولوژی باید به خوبی تعریف و مستند شده باشد و مستند آن باید توصیفی جامع، شفاف، منطقی، دقیق، تفصیلی و سازگار ارائه کند.

چرخه عمر و واحدهای کاری:

متدولوژی *DSDM*: متدولوژی دارای ۳ فاز است: ۱. Pre project. ۲. Project proper. ۳. Post project و به صورت ترتیبی اجرا می شود. خود فاز project proper دارای ۴ زیر فاز است. ۱. Feasibility. ۲. Foundation. ۳. Evolutionary Development. ۴. Deployment که زیر فاز های ۱ و ۲ به صورت ترتیبی و فاز های ۳ و ۴ آن تکراری اجرا می شوند و چرخه عمر آن هم کامل است.

متدولوژی *Crystal Clear*: متدولوژی دارای ۳ فاز است: ۱. Chartering. ۲. Cyclic delivery. ۳. Wrap-up که پشت سر هم اجرا می شوند و فرآیند ایجاد به صورت تکراری در فاز cyclic delivery انجام می پذیرد ولی فاز نگهداری را ندارد و چرخه کامل را پوشش نمی دهد.

مقایسه *DSDM* با *Crystal Clear*: هر دو به خوبی چرخه عمر و واحدهای کاری را تعریف کرده اند ولی چون *DSDM* فاز نگهداری دارد و چرخه کامل را پوشش می دهد بهتر است.

نقش ها و افراد:

متدولوژی *DSDM*: متدولوژی نقش ها به خوبی تعریف کرده و چهار دسته هستند. نقش هایی که مسئولیت دارند، نقش هایی که پاسخ گو هستند، نقش هایی که مورد مشورت قرار می گیرند و نقش هایی که به آنها

اطلاعات داده می شود. همچنین نقش ها دسته بندی دیگری هم دارند: نقش های کسب و کاری، نقش های مدیریتی، نقش های فرآیندی و نقش های فنی. نقش های آن عبارتند از: Business sponsor، Business، Visionary، Technical Coordinator، Business Analyst، Project Manager، Team، Leader، Business Ambassador، Solution Developer، Solution Tester، Technical، Advisor، Business Advisor، DSDM Coach، Workshop Facilitator و همچنین نقش های دیگر هم در صورت لزوم می توانند اضافه شوند.

متدولوژی *Crystal Clear*. در این متدولوژی یک Executive Spenser داریم که پول می دهد و پشتیبانی لجستیک دارد. Ambassador User وجود دارد که هر روز همراه تیم است و نیازمندی ها را تعریف و تأییدیه می دهد. Lead Designer سر تیم است و به عنوان مدیر پروژه عمل و خودش خبره فنی هم هست. نقش های دیگری هم هستند مثل System Analyst، Designer Programmer، Business Expert، Tester، Text Writer، Coordinator و نقش های دیگر.

مقایسه *DSDM* با *Crystal Clear*. متدولوژی *DSDM* در این معیار بهتر است چون نقش های بیشتری دارد و آنها را دقیق تر تعریف کرده است و برای فعالیت های بیشتری نقش دارد.

زبان مدل سازی:

متدولوژی *DSDM*. متدولوژی مدل سازی دارد و تأکید زیاد هم روی آن دارد ولی روی انتخاب زبان مدل سازی تأکید ندارد.

متدولوژی *Crystal Clear*. درباره زبان مدل سازی در این متدولوژی اجبار به استفاده از زبان خاصی نشده است اما از UML به عنوان زبانی که می توان با آن مدل سازی انجام داد نام برده است.

مقایسه *DSDM* با *Crystal Clear*. هر دو روی مدل سازی تأکید دارند و اجبار به استفاده از زبان خاصی نکرده اند و در این معیار مشابه هستند.

محصولات:

متدولوژی *DSDM*. در این متدولوژی محصولات به خوبی تعریف شده اند. محصولات آن شامل این موارد است: Terms of Reference، توجیه اقتصادی، لیست اولویت بندی شده نیازمندی ها، تعریف راه حل معماری، تعریف رویکرد ایجاد، برنامه تحویل، تعریف رویکرد مدیریتی، ارزیابی امکان پذیری، Foundation

Summary، موارد پشتیبانی ها، نمونه سازی ها، مدل ها، تست ها و تضمین کیفیت، ارزیابی سود، برنامه Timebox، رکورد های بازبینی Timebox، گزارش بازبینی پروژه

متدولوژی *Crystal Clear*. محصولات نیاز و نقش های مسئول به خوبی تعریف شده اند و محصولات شامل این موارد است: بیانیه ماموریت همراه با اولویت بندی Trade-off ها، ساختار تیم و قراردادهای، نتایج reflection workshop، نقشه پروژه، برنامه عرضه، وضعیت پروژه، فهرست ریسک ها، برنامه تکرار و وضعیت آن، برنامه زمان بندی، فهرست اهداف عامل ها، usecase و نیازمندی ها، مدل نقش کاربر، توصیف معماری، پیش نویس های نمایش، مدل قلمرو عمومی، یادداشت ها و طرح های طراحی، کد منبع، کد مهاجرت، تست ها، بسته بندی سیستم، گزارش باگ ها و راهنمای کاربری

مقایسه *DSDM* با *Crystal Clear*. هر دو متدولوژی محصولات را به خوبی تعریف کرده اند و تقریباً مشابه هم هستند.

تکنیک ها و قواعد:

متدولوژی *DSDM*. تکنیک ها و قواعد و اصول متدولوژی به خوبی تعریف شده اند. بعضی روی ها و تکنیک ها در *DSDM* اجباری هستند.

متدولوژی *Crystal Clear*. تکنیک ها و قواعد برای انجام فعالیت ها را به خوبی تعریف کرده است و برای هر فعالیت تکنیک های مختلفی را حتی از متدولوژی های دیگر برای استفاده معرفی کرده است.

مقایسه *DSDM* با *Crystal Clear*. در این معیار *DSDM* تکنیک هایی را اجبار می کند که از این لحاظ بهتر است همچنین تکنیک ها و قواعد دقیق تری دارد ولی *Crystal Clear* تکنیک ها را به صورت کلی برای اعمال توصیه کرده است.

فعالیت های چتری:

متدولوژی *DSDM*. توجه به فعالیت های چتری خوبی است. به مدیریت پروژه توجه دارد. برنامه ریزی در آن خوب است. توجه زیادی به کیفیت دارد و چون فرآیند های سازمانی دارد توجه مدیریت ریسک هم در آن خوب است.

متدولوژی *Crystal Clear* فعالیت های چتری را پوشش می دهد و در سراسر آن برنامه ریزی وجود دارد و فعالیت هایی انجام می شود که مدیریت ریسک می کنند همچنین تکراری افزایش بودن آن کمک به مدیریت ریسک می کند. همچنین **Test Base Development** بودن باعث کمک به تضمین کیفیت آن است.

مقایسه *DSDM* با *Crystal Clear*. هر دو متدولوژی پوشش فعالیت های چتری را دارند ولی *DSDM* به دلیل وجود فرآیند های سازمانی و نیاز به اعمال مدیریت از طرف کسب و کار در فعالیت های چتری دقیق تر و بهتر است.

معیار پوشش چرخه عمر عمومی ایجاد نرم افزار:

فعالیت های چرخه عمر عمومی شامل تعریف، ایجاد و نگهداری هستند که اگر این سه مرحله کامل پوشش داده شوند چرخه عمر کامل است.

تعریف:

کاوش قلمرو مسئله و مدل سازی آن:

متدولوژی *DSDM* در زیر فاز **Feasibility** قلمرو مسئله کاوش می شود و چيستی سیستم بدست می آید.

متدولوژی *Crystal Clear* کاوش قلمرو مسئله در فاز **Chartering** و در فعالیت ۳۶۰ درجه کاوشی انجام می شود. چيستی سیستم در می آید و قلمرو مسئله می تواند با کلاس دیاگرام مدل شود.

مقایسه *DSDM* با *Crystal Clear*. در این معیار تقریباً مشابه هستند و کاوش قلمرو مسئله را انجام می دهند.

استخراج نیازمندی ها:

متدولوژی *DSDM* در زیر فاز **Foundation** مجموعه اولویت بندی شده نیازمندی ها در می آید که البته به مرور می تواند کامل شود. جزئیات نیازمندی ها در تکرار ها بدست می آید.

متدولوژی *Crystal Clear* استخراج نیازمندی ها در فاز **Chartering** و در فعالیت ۳۶۰ درجه کاوشی انجام می شود. که در قالب **use case** استخراج می شوند.

مقایسه *DSDM* با *Crystal Clear*. در این معیار هر دو متدولوژی مشابه هستند.

امکان سنجی:

متدولوژی *DSDM* در زیر فاز **Feasibility** امکان سنجی انجام می شود که شامل امکان سنجی فنی و مالی می باشد.

متدولوژی *Crystal Clear* امکان سنجی در فاز **Chartering** و در فعالیت ۳۶۰ درجه کاوشی انجام می شود و یک توجیه اقتصادی هم برای پروژه در می آید.

مقایسه *DSDM* با *Crystal Clear* در این معیار چون در **DSDM** اعمال مدیریت از سمت سازمان امکان سنجی بسیار دقیق تر و بهتر است.

ایجاد:

طراحی معماری:

متدولوژی *DSDM* در فاز **Foundation** معماری طراحی می شود و در **solution architecture** قرار می گیرد که هم معماری نرم افزار و هم معماری سازمان است.

متدولوژی *Crystal Clear* معماری در فاز **Chartering** و در فعالیت ۳۶۰ درجه کاوشی انجام می شود و با مقایسه معماری های مختلف آن که شدنی است انتخاب می شود. خود معماری اولیه در تکرار اول ساخته، برنامه نویسی شده و تست می شود و تکمیل، گسترش و تکامل آن در تکرارهای بعدی است.

مقایسه *DSDM* با *Crystal Clear* در این معیار **DSDM** چون معماری در سطح سازمان را هم دارد قوی تر است.

طراحی تفصیلی:

متدولوژی *DSDM* طراحی تفصیلی به خوبی در فازهای تکراری انجام و جزئیات نیازمندی ها در می آید و مدل سازی انجام می شود.

متدولوژی *Crystal Clear* فعالیت های طراحی را تشویق می کند که به عنوان **design note** مستند می شوند. ولی به طور کلی طراحی توسط خود برنامه نویسان که **designer programmer** هستند انجام و محصولات آن به اشتراک گذاشته نمی شود.

مقایسه *DSDM* با *Crystal Clear* از لحاظ طراحی تفصیلی **DSDM** بهتر است و تاکید بیشتری روی آن دارد.

پیاده سازی:

متدولوژی *DSDM* در زیر فاز **Evolutionary development** پیاده سازی به صورت تکاملی انجام می شود.

متدولوژی *Crystal Clear* پیاده سازی در قالب تکرار ها در فاز **cyclic delivery** و فعالیت **develop in iteration** انجام می شود.

مقایسه *DSDM* با *Crystal Clear* در این معیار هر دو مشابه هستند.

تست:

متدولوژی *DSDM* تست همراه با پیاده سازی وجود دارد و به صورت **Test Base Development** است و روی تست خیلی تاکید است. تست های سیستمی هم در آن به خوبی انجام می شود.

متدولوژی *Crystal Clear* این متدولوژی **Test Base Development** است و تست با پیاده سازی است همچنین بعد از **integration** تست انجام می شود و در فاز **wrap up** هم **acceptance test** داریم.

مقایسه *DSDM* با *Crystal Clear* در هر دو تست ها به خوبی انجام می شوند اما به دلیل اینکه پروژه ها در **DSDM** می تواند بزرگتر و در سطح سازمان ها باشد تست مهم تر و تاکید بیشتری روی آن است.

استقرار:

متدولوژی *DSDM* استقرار در زیر فاز **Deployment** انجام و **Solution Increment** ها با هم **assemble** شده و استقرار می یابند.

متدولوژی *Crystal Clear* بعد از بازخورد نهایی کاربران بعد از **deliver to user** در فاز **cyclic delivery** استقرار انجام می شود. همچنین باقی مانده موارد استقرار نیافته در فاز **wrap up** انجام خواهد شد.

مقایسه *DSDM* با *Crystal Clear* فعالیت های مربوط به استقرار در **DSDM** دقیق تر و با جزئیات و فعالیت های بیشتر و دقیق تری هستند و **DSDM** در این معیار بهتر است.

نگهداری:

متدولوژی *DSDM* نگهداری پس از اتمام عرضه ها در فاز **post project** انجام می پذیرد.

متدولوژی *Crystal Clear* فعالیتی به عنوان نگهداری وجود ندارد.

مقایسه DSDM با Crystal Clear. متدولوژی DSDM چون پوشش نگهداری را دارد بهتر است.

معيار پشتیبانی از فعالیت های چتری:

متدولوژی باید فعالیت های چتری را پوشش بدهد. فعالیت های چتری فعالیت هایی هستند که در همه بخش های چرخه فرایند لحاظ می شوند و شامل مدیریت ریسک از طریق ارزیابی ریسک و فعالیت های کاهش ریسک گنجانده شده در چرخه عمر، مدیریت پروژه از طریق برنامه ریزی، زمان بندی و تکنیک های کنترلی قرار گرفته در فرایند و تضمین کیفیت با ارزیابی کیفیت و تکنیک های بهبود کیفیت در فرایند هستند.

مدیریت ریسک:

متدولوژی DSDM. دموهای مداوم جهت آگاه سازی و نمود اعمال کنترل وی باعث کاهش ریسک می شود. نمونه سازی به دقیق کردن تخمین ها و کاهش ریسک نیازمندی کمک می کند که زیاد هم انجام می شود. تکراری افزایشی بودن که نهادینه است، عرضه زود هنگام و امکان سنجی اولیه هم تا حد خوبی به مدیریت ریسک کمک می کند. درآوردن معماری و همچنین نمونه سازی ها که در این متدولوژی انجام می شود می تواند به کاهش ریسک کمک کند. تاکید روی ساخت افزایشی بر پایه معماری در اصل ۵ کمک به مدیریت ریسک می کند.

متدولوژی Crystal Clear. تکراری افزایشی بودن که نهادینه است نوعی مدیریت ریسک دارد. همچنین استقرار مداوم، عرضه زود هنگام، امکان سنجی و درآوردن معماری باعث کاهش ریسک است. نمونه سازی ها که در این متدولوژی انجام می شود می تواند به کاهش ریسک کمک کند. کنار هم بودن افراد به هم باعث کاهش ریسک می شود.

مقایسه DSDM با Crystal Clear. در این معیار DSDM فعالیت های بیشتری دارد و قوی تر است و به دلیل وجود اعمال مدیریت سازمان بهتر از Crystal Clear می باشد.

مدیریت پروژه:

متدولوژی DSDM. در delivery plan برنامه ریزی انجام می شود و برای کل پروژه حدودی و برای فازهای بلافاصله بعد دقیق است که به مدیریت پروژه کمک می کند. همچنین Management approach definition وجود دارد که برای مدیریت پروژه است و در آن زمان بندی، روش بازنگری، پایش پیشرفت و ... مشخص می شود. Timebox بودن به زمان بندی کمک می کند. همچنین کنترل روی سازمان هم کمک

زیادی به مدیریت پروژه می کند. همچنین اصل ۲ آن مربوط به تحویل سر وقت است که رسیدن به آن نیازمند اعمال مدیریت روی پروژه است.

متدولوژی *Crystal Clear* در فاز **Chartering** و در فعالیت ۳۶۰ درجه کاوشی برنامه ریزی پروژه انجام می شود، انواع منابع مورد نیاز تعیین می شوند. یک **initial Project Plan** در می آید که شامل دو بخش است، یکی **Project map** که می گوید تکالیف ایجاد چه هستند و یکی هم **release plan** که زمان اتمام کار و اهداف تکرارها را مشخص می کند و برنامه ریزی هم تنها می توان برای **cyclic delivery** پیش رو باشد و نه کل آن. در فاز **cyclic delivery** و در فعالیت **recalibrate the release plan** نیازمندی و برنامه ریزی پروژه دیده و **release plan** هم اعتبار سنجی می شود. همچنین **Blitz programming** که برنامه ریزی تکالیف است هم انجام می شود.

مقایسه *DSDM* با *Crystal Clear*. در این معیار **DSDM** بهتر و اصول تاکید شده ای برای مدیریت پروژه دارد و فعالیت های سازمانی هم به بهتر شدن آن از **Crystal Clear** کمک می کند.

تضمین کیفیت:

متدولوژی *DSDM*. طبق اصل ۴ کیفیت بالای محصول یکی از اصول **DSDM** است. از تست به عنوان فاکتور های حیاتی موفقیت نام برده شده و در متدولوژی نهادینه است که به تضمین کیفیت کمک می کند. دمو های مداوم به مشتری هم به کیفیت کمک می کند. **Benefit assessment** که معمولا هر شش ماه یا یک سال انجام می شود و محقق ماندن اهداف را بررسی می کند به حفظ کیفیت کمک می کند. در استقرار هم در **review** استانداردهای کیفی مورد بررسی قرار می گیرند که گاهی با یک چک لیست ساده و گاهی با کارگاه می تواند باشد که به کیفیت کمک می کند. تاکید روی ساخت افزایشی بر پایه معماری در اصل ۵ کمک به مدیریت کیفیت می کند.

متدولوژی *Crystal Clear*. برای تضمین کیفیت **Test Base Development** بودن و تست های مداوم که انجام می شود همچنین تأییدیه کاربر و حضور دائمی در تیم و تست پذیرش نهایی به آن کمک می کند. در جلسات **reflection workshop** هم محصول مورد بازبینی قرار می گیرد و مسائل مربوط به کیفیت هم مدیریت می شود.

مقایسه *DSDM* با *Crystal Clear*. در این معیار **DSDM** بهتر است و فعالیت های بیشتری دارد و اعمال مدیریت سطح سازمان هم در بهتر بودن تضمین کیفیت مهم است.

معيار بی درزی و گذر هموار:

بی درزی به معنی عدم وجود تغییر الگو در زنجیره ی مدل سازی و فعالیت ها است. گذار بین فازها، مراحل و تکالیف باید هموار باشند و باید ادامه طبیعی هم باشند و میان آنها وقفه ایجاد نشود.

بی درزی و گذار هموار:

متدولوژی *DSDM* به دلیل نداشتن محصولات به کلی جدید و تکمیل محصولات و تکراری افزایشی بودن گذر آن هموار است اما مثل همه متدولوژی های شی گرا دارای درز است.

متدولوژی *Crystal Clear* این متدولوژی درز دارد و به آن توجهی نمی شود. به دلیل تکراری افزایشی بودن گذار بین فازها و فعالیت آن هموار است.

مقایسه *DSDM* با *Crystal Clear*. گذر ها در *DSDM* هموار است در *Crystal Clear* هم گذر هموار است و در هر دو گذار هموار است. از لحاظ بی درزی *Crystal Clear* توجهی به آن ندارد چون اصولاً زنجیره مدل سازی ندارد و *DSDM* با وجود درز ذاتی متدولوژی های شی گرا بهتر است.

معيار مبتنی بودن بر نیازمندی ها:

کارها باید مبتنی بر نیازمندی ها باشند. نیازمندی های وظیفه ای و غیر وظیفه ای باید در اوایل فرایند متدولوژی استخراج شوند، در جای درست خود مدل سازی شده و مبنای کارهای طراحی، پیاده سازی و تست قرار گیرند.

متدولوژی *DSDM* پیشنهاد استفاده از *user story* است و متدولوژی مبتنی بر نیازمندی هاست. همچنین با مبتنی بر نیازمندی ها بودن نیازمندی های غیر عملکردی هم مورد توجه است.

متدولوژی *Crystal Clear* بعد از شکستن *user story* ها به *task* ها همه چیز برحسب آنها پیش می رود و دیگر مبتنی بر نیازمندی ها نیست. نیازمندی های غیر وظیفه ای هم معمولاً در معماری مشخص می شوند.

مقایسه *DSDM* با *Crystal Clear*. در این معیار *DSDM* بهتر است چون بر نیازمندی ها مبتنی است.

معیار آزمون پذیری، ملموس بودن و قابل رهگیری بودن به نیازمندی ها:
مصنوعات باید آزمون پذیر، ملموس و قابل فهم و قابل نگاشت به نیازمندی ها باشند.

آزمون پذیری:

متدولوژی *DSDM* محصولات کم هستند و به مرور تکمیل می شوند که باعث می شود آزمون پذیری آن خوب باشد.

متدولوژی *Crystal Clear* مصنوعات آن کد است که به خوبی قابل آزمون است و مدل سازی ها هم چون توسط خود افراد انجام می شود و از هم جدا هستند به خوبی قابل آزمون هستند.

مقایسه *DSDM* با *Crystal Clear*. از لحاظ آزمون پذیری چون *Crystal Clear* محصولات اشتراکی ندارد این معیار خیلی موضوعیت ندارد و *DSDM* بهتر است. ولی اگر آزمون پذیری مصنوعات ساخته شده توسط خود افراد مد نظر باشد چون *Crystal Clear* محصولات کوچک تر و کمتری توسط خود هر فرد دارد از این لحاظ بهتر خواهد بود.

ملموس بودن:

متدولوژی *DSDM* محصولات هدفمند هستند و کاملاً برای کسانی که استفاده می کنند ملموس هستند.

متدولوژی *Crystal Clear* محصولات تولیدی برای خود برنامه نویس که آنها را تولید می کند کاملاً ملموس هستند. همچنین نمونه سازی هم کمک به ملموس شدن می کند. سایر محصولات هم برای استفاده کنندگان به خوبی ملموس هستند.

مقایسه *DSDM* با *Crystal Clear*. هر دو متدولوژی محصولاتی که تولید می کنند ملموس و برای استفاده کنندگان قابل فهم هستند.

قابل رهگیری به نیازمندی ها:

متدولوژی *DSDM* چون مبتنی بر نیازمندی هاست قابلیت رهگیری نیازمندی ها را دارد.

متدولوژی *Crystal Clear* چون بعد از شکستن نیازمندی ها به تکالیف دیگر مبتنی بر نیازمندی نیست در رهگیری سخته دارد و باید برای رهگیری تدابیر لازم اندیشیده شود ولی بالاخره چون باید تحقق نیازمندی ها را سنجید رهگیری باید انجام شود.

مقایسه *DSDM* با *Crystal Clear*. در *DSDM* این معیار بهتر است و *Crystal Clear* برای رهگیری سخته دارد.

معیار تشویق به مشارکت فعالانه کاربران:

کاربران باید به طور فعالانه در فرایند ایجاد نرم افزار دخالت داشته باشند که این کار در مدیریت مخاطرات و تضمین کیفیت اهمیت دارد.

متدولوژی *DSDM*. همکاری با مشتری از اصول *DSDM* است (اصل ۳). ارتباط شفاف و مداوم جهت بیان پیشرفت ها و مشکلات که با دمو دادن محقق می شود هم از اصول *DSDM* است. (اصل ۷). تحویل های زود به زود و دریافت بازخورد و اعمال مدیریت از سوی سازمان روی پروژه هم نوعی مشارکت کاربران است.

متدولوژی *Crystal Clear*. در این متدولوژی نقش *Ambassador User* وجود دارد که تعریف نیازمندی ها و تأییدیه کار را انجام می دهد. همچنین در *deliver to real user* به تعداد محدود کاربر نشان داده و از آنها بازخورد گرفته می شود.

مقایسه *DSDM* با *Crystal Clear*. در این معیار با توجه به تاکید بیشتر *DSDM* بهتر است و به دلیل اینکه باید اعمال مدیریت سازمان باشد سازمان هم بیشتر در کار درگیر است.

معیار قابل اجرا بودن و کارا بودن:

باید بتوان فرایند را به راحتی اجرا کرد و در ایجاد نرم افزار به کار گرفت همچنین باید بتوان از فرایند به طور کارا، موثر و مفید در حوزه کاربرد استفاده کرد.

قابل اجرا بودن:

متدولوژی *DSDM*. فرآیند را می توان به خوبی اجرا کرد و *suitable filter* های آن هم باعث می شود قابل اجرا باشد. مبتنی بر معماری بودن هم قابلیت اجرا را بالا می برد.

متدولوژی *Crystal Clear*. فرآیند کمی سنگین است که می تواند قابل اجرا بودن را کاهش دهد. همچنین فاز نگهداری را ندارد که آن هم می تواند باعث کم شدن قابلیت اجرا شود. با کمک ابزارها می توان فرآیند را خوبی اجرا کرد که قابلیت کارایی را افزایش می دهد. توجه به معماری قابلیت اجرا را بالا می برد.

مقایسه *DSDM* با *Crystal Clear*. از لحاظ قابل اجرا بودن *DSDM* بهتر عمل می کند و می تواند قابل اجرا بودن را از همان ابتدای کار مشخص کند.

کارا بودن:

متدولوژی *DSDM*. پیچیدگی کم، عدم وابستگی به ابزار خاص و فعالیت های قوی مدیریت پروژه کمک می کند کارا بتواند آن را اجرا کرد. این که مدل سازی اشتراکی دارد هم می تواند باعث کارا بودن آن شود.

متدولوژی *Crystal Clear*. مبتنی بر ابزار بودن برای اجرای برخی فعالیت ها اجرای کارا را کاهش می دهد. البته توجه به معماری باعث می شود بتون آن را کارا اجرا نمود. فعالیت های برنامه ریزی در آن هم می تواند به خوبی به کارا بودن آن کمک کنند. البته تکیه به ارتباطات انسانی و اجبار به کنار هم بودن تیم ها و عدم پشتیبانی تیم های توزیع شده می تواند به کارا بودن اجرا آسیب وارد کند. همچنین نداشتن مدل سازی سیستماتیک به کارا اجرا کردن آن ضربه می زند.

مقایسه *DSDM* با *Crystal Clear*. از این لحاظ *DSDM* بهتر است چون مواردی که باعث اجرای کارا می شود در آن قوی تر هستند.

معیار قابل مدیریت بودن پیچیدگی:

فرایند از تعدادی واحد کاری، نقش، محصول و تعدادی معیار سنجش تشکیل می شود. باید خود این مجموعه از نظر پیچیدگی قابل مدیریت باشد. مدیریت معمولاً به دو روش بخش بندی و لایه بندی انجام می شود.

متدولوژی *DSDM*. برای مدیریت پیچیدگی ها ۳ فاز اصلی دارد. همچنین فاز ۲ هم ۴ زیرفاز دارد. که اینها بخش بندی است. همچنین در فاز ۲، ۲ فاز ترتیبی و ۲ فاز تکراری هستند. که فاز های تکرار آن لایه بندی هستند.

متدولوژی *Crystal Clear*. فرآیند برای مدیریت پیچیدگی کل آن را به فاز تقسیم که بخش بندی است. همچنین تکرار هایی برای ایجاد هم که در فاز دو انجام می شوند لایه بندی هستند همچنین در تکرار ها هم *episode* وجود دارد. شکستن نیازمندی ها به تکالیف هم می تواند به مدیریت پیچیدگی کمک کند.

مقایسه *DSDM* با *Crystal Clear*. در این معیار هر دو متدولوژی پیچیدگی ها را به خوبی مدیریت می کنند و مشابه هستند.

معیار قابلیت گسترش، مقیاس پذیری، پیکربندی و انعطاف پذیری:

با بتوان فرایند یک متدولوژی را با کمک نقاط گسترش و مکانیزم های به صراحت تعریف شده آن گسترش داد، در پروژه های با اندازه مختلف و سطوح بحرانیت مختلف اعمال کرد، در ابتدای پروژه و مناسب موقعیت فعلی آن را پیکربندی نمود و بتوان در جریان اجرای فرایند پیکربندی را تغییر داد و اصلاح کرد.

گسترش پذیری:

متدولوژی *DSDM* صحبتی در مورد گسترش پذیری آن نشده است.

متدولوژی *Crystal Clear* در فاز *Chartering* و در فعالیت ۳۶۰ درجه کاوشی می توان درجات آزادی را تعیین، ابزارها را تعیین کرد و خیلی چیزهای دیگر در اینجا می توانند انجام بشوند. هر متدولوژی را می توان با اضافه کردن محصولات جدید گسترش داد و ان را بهبود بخشید که باعث افزایش مقیاس پذیری هم می شود.

مقایسه *DSDM* با *Crystal Clear* در این معیار *Crystal Clear* بهتر است و *DSDM* صحبتی در مورد گسترش پذیری ندارد.

مقیاس پذیری:

متدولوژی *DSDM* مقیاس پذیری آن خوب است چون مدل سازی دارد. در مورد هماهنگی تیم رویه دارد و ارزیابی ریسک به مقیاس پذیری کمک می کند و نداشتن فرمالیزم هم باعث می شود مقیاس پذیری اش کاهش یابد. البته خودش می گوید مناسب سیستم های بزرگ نیست.

متدولوژی *Crystal Clear* مقیاس پذیری خیلی محدودی دارد و *Crystal Clear* مخصوص نوع خاصی از پروژه ها یعنی *C6* و *D6* است که مربوط به بحرانیت *Comfort* و *Discretionary Money* با شش نفر عضو تیم است.

مقایسه *DSDM* با *Crystal Clear* در این معیار *DSDM* مقیاس پذیرتر است و می تواند پروژه های بزرگتری را به دلیل اینکه می تواند چندین تیم را پشتیبانی کند پوشش بدهد ولی هیچکدام سطح بحرانیت *Life* را پوشش نمی دهند.

پیکربندی:

متدولوژی *DSDM* متدولوژی قابل پیکر بندی است در فاز **Foundation** می توان آن را پیکربندی کرد و نحوه اجرا، محیط اجرا ، نحوه مدیریت پروژه، نحوه تست و ... را تعیین نمود و **tailoring** انجام می شود. همچنین پرسش نامه **PAQ** و فیلترهای مطلوبیت هم به پیکربندی کمک می کند.

متدولوژی *Crystal Clear* در فاز **Chartering** و در فعالیت ۳۶۰ درجه کاوشی می توان درجات آزادی را تعیین، ابزارها را تعیین کرد و خیلی چیزهای دیگر در اینجا می توانند انجام بشوند. خود انتخاب **Crystal Clear** از میان سایر متدولوژی ها هم اینجا انجام می شود.

مقایسه *DSDM* با *Crystal Clear*. از این لحاظ *DSDM* امکانات و معیار های بیشتری را برای پیکربندی فراهم می کند و بهتر است و با توجه به اینکه خود **Crystal Clear** نتیجه پیکربندی **Crystal** است در این معیار *DSDM* بهتر است.

انعطاف پذیری:

متدولوژی *DSDM* پس از **review** در استقرار **retrospective** انجام و محصول، فرایند، برنامه و اصول مورد بازبینی و تنظیم قرار می گیرند که می توان تغییرات را در طول اجرای پروژه اعمال نمود.

متدولوژی *Crystal Clear* در **recalibrate the release plan** برنامه مورد بازنگری قرار می گیرد. بعد از تحویل به کاربر در **reflect on deliver** هر سه محصول ، برنامه و فرآیند مورد بازنگری قرار می گیرد و در صورت لزوم تغییر می کند. در پایان فاز هم **reflective workshop** برگزار و همه چیز بررسی خواهد شد.

مقایسه *DSDM* با *Crystal Clear*. در این معیار هر دو متدولوژی انواع بازنگری و کارگاه دارند و مشابه هم هستند.

معیار حوزه کاربرد:

متدولوژی در چه حوزه هایی کاربرد دارد و برای چه حوزه هایی مناسب نیست حداقل باید بتواند حوزه کاربرد سیستم های اطلاعاتی را پوشش دهد.

متدولوژی *DSDM* قابلیت کاربرد محدود به سیستم های تجاری است و سیستم های **Life Critical** را پشتیبانی نمی کند.

متدولوژی *Crystal Clear*. متدولوژی طیف وسیعی را پوشش نمی دهد و تنها سطح بحرانیت **Comfort** و **Discretionary Money** را پوشش می دهد.

مقایسه *DSDM* با *Crystal Clear*. در این معیار **DSDM** بهتر است و می تواند پروژه های بزرگتر را پوشش بدهد ولی هر دو نهایتاً مناسب سیستم های اطلاعاتی هستند ولی **DSDM** چون سطوح بحرانیت بیشتری را پوشش می دهد و مناسب فعالیت های سازمانی هم دارد حوزه کاربردش بیش از *Crystal Clear* است.

معیارهای زبان مدل سازی:

معیار پشتیبانی از مدل سازی شی گرا سازگار، دقیق و بدون ابهام:

مدل ها باید با کیفیت و شامل هر چه در یک مدل شی گرا نیاز است باشد و بتوان به وضوح شی گرایی را در آن دید، مدل ها نباید یکدیگر را نقض کنند، مدل باید صحیح باشند و اشتباه نداشته باشند همچنین باید با جزئیات کامل باشد و زبان مدل سازی اینها را پشتیبانی کند.

دیدگاه های مختلف مدل سازی:

متدولوژی *DSDM*. مدل سازی ساختاری و وظیفه ای روی قلمرو مسئله و سیستم را دارد.

متدولوژی *Crystal Clear*. تنها برای معماری کلاس دیاگرام رسم می کند ولی چون مدل سازی سیستماتیک نیست و بر عهده خود افراد است نمی توان در باره آن خیلی بحث کرد.

مقایسه *DSDM* با *Crystal Clear*. در این معیار چون **DSDM** خیلی روی مدل سازی تاکید دارد و مدل سازی سیستماتیک است بهتر است.

مدل سازی از منطقی به فیزیکی:

متدولوژی *DSDM*. مدل سازی از منطقی به فیزیکی را دارد و محصولات از فاز **pre project** که در قلمرو مسئله است شروع می شود و با تکمیل به کد ختم می شود.

متدولوژی *Crystal Clear*. به دلیل اینکه مدل های اشتراک گذاشته نمی شوند خیلی این مورد مورد توجه نیست.

مقایسه *DSDM* با *Crystal Clear*. در این معیار **DSDM** بهتر است.

سطوح مختلف انتزاع و دانه بندی:

متدولوژی *DSDM* مدل سازی را از قلمرو مسئله شروع می کند و تا کلاس ها ادامه می دهد تا بتوان کد را از روی آن زد.

متدولوژی *Crystal Clear* چون مدل سازی سیستماتیک ندارد خیلی به این موضوع توجه نمی شود و افراد خودشان ممکن است سطوح مختلف را مدل کنند ولی به اشتراک گذاشته نمی شود.

مقایسه *DSDM* با *Crystal Clear* در این معیار چون مدل سازی در *DSDM* سیستماتیک است بهتر است و می تواند سطوح مختلف را پوشش بدهد.

مدل سازی صوری و غیر صوری:

متدولوژی *DSDM* مدل سازی دارد ولی از فرمالیزم پشتیبانی نمی کند.

متدولوژی *Crystal Clear* از فرمالیزم پشتیبانی نمی کند.

مقایسه *DSDM* با *Crystal Clear* از لحاظ مدل سازی *DSDM* بهتر است چون مدل سازی سیستماتیک است ولی هیچکدام فرمالیزم را پشتیبانی نمی کنند که مشابه هم هستند.

معیار مدیریت تناقض ها و پیچیدگی ها:

پیچیدگی مدل ها باید مدیریت شود و از بوجود آمدن ناسازگاری بین آنها جلوگیری نمود. زبان مدل سازی باید برای تشخیص و رفع ناسازگاری و پیچیدگی مدل ها استراتژی و تکنیک های مناسب داشته باشد.

متدولوژی *DSDM* برای مدیریت تناقض ها و پیچیدگی ها فعالیت های قوی مدیریتی دارد و چون به کیفیت خیلی اهمیت می دهد می تواند تناقض ها و پیچیدگی های مدل ها را خوب مدیریت کند. این که تاکید روی مدل سازی به اندازه کافی دارد می تواند باعث کم شدن پیچیدگی و ناسازگاری شود.

متدولوژی *Crystal Clear* چون مدل سازی سیستماتیک نیست و بر عهده خود تیم است توجهی به این موضوع نمی شود. البته چون یک نفر مدل سای می کند معمولا در مدل هایش تناقض و پیچیدگی ها کم هستند.

مقایسه *DSDM* با *Crystal Clear* در معیار امکان مدیریت تناقض ها و پیچیدگی ها در *DSDM* وجود دارد و بهتر است ولی در *Crystal Clear* چون مدل سازی سیستماتیک نیست خیلی این مورد موضوعیت ندارد.